Deployment Manual

CLC Workbenches

Manual for

CLC Workbenches: deployment and technical information, version 1.6

Windows, Mac OS X and Linux

August 8, 2012

This software is for research purposes only.

CLC bio
Finlandsgade 10-12
DK-8200 Aarhus N
Denmark

Contents

1 Introduction **5**

1.1 Deployment strategies 5

2 Installation **7**

2.1 System requirements 7

2.2 Available installers 8

2.2.1 Java 8

Java on Mac 9

2.2.2 Overview of available installers 9

2.3 What does it do? 9

2.3.1 Extracting and copying files to the installation directory 9

2.3.2 Setting the amount of memory available 10

2.3.3 Shortcuts and file associations 10

2.4 Silent installation 10

3 License **11**

3.1 License server set-up on clients 11

4 Plug-ins and resources **13**

5 Workflows **14**

6 Connecting to a CLC Server **15**

7 Security policies **16**

8 Storing and backing up data **18**

8.1 Storing data 18

8.1.1 Data structure 18
8.1.2 Changing the default location 19
8.2 Back-up of data 19
8.3 Special configurations for large amounts of data 20
8.3.1 Temporary data 20
8.3.2 Disk space requirements 20

9 System resources **22**
9.1 Setting the amount of memory available for the JVM 22
9.2 Setting the number of cores to use 23

10 Overview - where do we put things? **24**
10.1 Computer-level information 24
10.2 User-level information 25

Index **25**

Chapter 1

Introduction

If you are in charge of installing and maintaining CLC Workbenches in your organization, you probably have a lot of questions about installers, licenses, and where do we put files for this and that.

This manual is written to answer these questions. It is primarily aimed towards client installations (i.e. the CLC Workbenches). For installing the *CLC Bioinformatics Database* and *CLC Server*, please refer to the installation chapter in the respective user manuals (see http://www.clcbio.com/usermanuals).

At the moment, the CLC Workbenches are:

- *CLC Main Workbench*
- *CLC Genomics Workbench*

In addition, there is the *CLC Sequence Viewer* which is very similar to the other Workbenches except when it comes to licensing (it is free) and plug-ins (some plug-ins cannot be installed in the *CLC Sequence Viewer*).

In the first part of this manual, we take a closer look at the installer: what does it do, what are the options etc. Then we go into details of the license system, followed by an explanation of the concept of plug-ins. Finally there is an overview chapter which tells you where to find all the different files in a client installation. This is useful when deploying the Workbench but also for back-up purposes.

We will not go into details of Linux installations, but if you need these details, please contact us.

Questions, comments and feedback on this manual are very welcome at support@clcbio.com.

1.1 Deployment strategies

Deployment strategies should be developed so that they fit your existing IT set-up. However, the following solution has proven to work in many cases. Note that this is a brief summary of the steps to take, and that the rest of this manual is devoted to more detailed information.

1. On a single computer with the same OS as the target computers, complete a full installation including:

- Running the installer
- Set up license server connection [1]
- Install plug-ins and resources
- Configure security policies, path for temporary data and default location (see more information in this manual)
- Specify proxy server information if needed

2. Run the installer in silent mode (`-q`) on all the target computers
3. Copy the following files from the installation in 1) to the target computers
   - All files from the `settings` folder in the installation directory
   - `plugins` and `resource` folder from the installation directory

We recommend you tailor this strategy to your own organization - this is just an example of how it can be done.

Creating a GHOST image or similar to copy to all the target computers is also a possibility, but there are a few routines performed by the installer that need to be taken into acount, especially allocating memory (see section 9.1) and the creation of shortcuts and file associations.

[1] If you do not use a license server, you will have to activate licenses on each computer

Chapter 2

Installation

This chapter deals with the installer and related information about the installation process.

2.1 System requirements

The system requirements of the CLC workbenches except the *CLC Genomics Workbench* are these:

- Windows XP, Windows Vista, or Windows 7, Windows Server 2003 or Windows Server 2008
- Mac OS X 10.6 or later. Intel CPU required. However, Mac OS X 10.5.8 is supported on 64-bit Intel systems.
- Linux: RedHat 5 or later. SuSE 10 or later.
- 32 or 64 bit
- 256 MB RAM required
- 512 MB RAM recommended
- 1024 x 768 display recommended

The requirements for the *CLC Genomics Workbench* are:

- Windows XP, Windows Vista, or Windows 7, Windows Server 2003 or Windows Server 2008
- Mac OS X 10.6 or later. Intel CPU required. However, Mac OS X 10.5.8 is supported on 64-bit Intel systems.
- Linux: RedHat 5 or later. SuSE 10 or later.
- 1024 x 768 display recommended
- Intel or AMD CPU required
- Memory: The numbers below give minimum and recommended amounts for systems running mapping and analysis tasks. The requirements suggested are based on the genome size.

  - **E. coli K12 (4.6 megabases)**
    * Minimum: 2Gb RAM
    * Recommended: 4Gb RAM
  - **C. elegans (100 megabases)** and **Arabidopsis thaliana (120 megabases)**
    * Minimum: 4Gb RAM
    * Recommended: 8Gb RAM
  - **Zebrafish (1.5 gigabases)**
    * Minimum: 8Gb RAM
    * Recommended: 16Gb RAM
  - **Human (3.2 gigabases)** and **Mouse (2.7 gigabases)**
    * Minimum: 24Gb RAM
    * Recommended: 48Gb RAM

- **Special requirements for de novo assembly**. De novo assembly may need more memory than stated above - this depends both on the number of reads and the complexity and size of the genome. See http://www.clcbio.com/white-paper for examples of the memory usage of various data sets.
- 64 bit computer and operating system required to use more than 2GB RAM

2.2 Available installers

There are installers available for each platform (Windows, Mac OS X and Linux). Each of these installers is available in a 32-bit and a 64-bit version, except for the Mac OS X installer. For Linux, there is both a `.sh` installer and an `.rpm` package.

2.2.1 Java

The Workbenches are based on Java, and this means that there has to be a Java Runtime Environment (JRE) on the computer to run the Workbench. For both Linux and Windows, the installers have a built-in JRE that will be installed in the installation directory of the Workbench. The advantage of this is twofold:

1. For computers who do not already have a JRE installed, the need for downloading and installing a JRE is eliminated.
2. for computers who already have a JRE installed, there will never be compatibility problems because the Workbench always uses its own JRE.

The built-in JRE is the latest Java 6 JRE from Sun Microsystems (http://java.sun.com).

The JRE used for running the CLC Workbench will not interfere with existing JREs on the computer.

Java on Mac

Since the Workbench uses Apple's JRE, there is no JRE included in the installer. When running the Workbench on 64-bit systems, please make sure that the 64-bit Java is used for launching applications:

Go to `/Applications/Utilties/Java` and double click on Java Preferences. In the **Java application versions**, reorder the list to have JRE6/64 bit at the top. Note that this may change the behavior of other Java-based programs on the computer.

Note! After setting the Java preference, you can either uninstall and reinstall the workbench and the memory will be adjusted automatically, or you can set the memory yourself (see section 9.1).

2.2.2 Overview of available installers

The table below shows an overview of the installers that are available.

Platform
Windows (2000, XP , Vista and 7)
Mac OS X 10.6 or later. Intel CPU required. However, Mac OS X 10.5.8 is supported on 64-bit Intel systems.
Linux installer
Linux package

2.3 What does it do?

The installer performs the following tasks:

2.3.1 Extracting and copying files to the installation directory

The Workbench is installed into the following directory per default (we use *CLC Main Workbench 5* as example):

Windows C:\Program files\Main Workbench 5

Mac OS X Applications/Main Workbench 5

Note that each major version of a Workbench has its own installation directory. This means that when upgrading from e.g. *CLC Main Workbench 5* to *CLC Main Workbench 6*, the old installation directory of version 4 will be left untouched when you install *CLC Main Workbench 6*.

If you wish to remove the old installation, please run the Uninstall program.

Minor updates will use the existing installation directory of the Workbench.

The installation directory can be defined during installation - the above are the default installation directories (see section 2.4 for more information on how to define the installation directory).

2.3.2 Setting the amount of memory available

The installer investigates the amount of RAM during installation and sets the amount of memory that the Workbench can use. Read more in section 9.1.

2.3.3 Shortcuts and file associations

The installer also creates shortcuts for starting the Workbench, and it creates file associations so that `.clc` files will be opened by the Workbench.

2.4 Silent installation

The installer also has a silent installation mode which is activated by the `-q` parameter when running the installer from a command line, e.g.

```
CLCMainWorkbench_5_6.exe -q
```

On Windows, if you wish to have console output, `-console` can be appended *as the second parameter* (this is only needed when running on Windows where there is no output per default):

```
CLCMainWorkbench_5_6.exe -q -console
```

You can also in silent mode define a different installation directory: `-dir`.

```
CLCMainWorkbench_5_6.exe -q -console -dir "c:\bioinformatics\clc"
```

Note! Both the `-console` and the `-dir` options only work when the installer is run in silent mode.

The `-q` and the `-console` options work for the Uninstall program as well.

Chapter 3

License

There are fundamentally two kinds of licenses for the Workbenches:

Fixed license A license order ID has to be activated against our server for each computer. The license will then be fixed to this computer. This requires manual intervention for each activation.

Floating license A license server is installed in your organization. It hosts a number of licenses which can be shared among all computers. Note that the license server is available for both Linux, Windows and Mac OS X.

For large installations, the floating license is by far the best option, since all the license administration takes place on the server (find the manual for the license server together with the server distribution). The fixed license requires manual work during installation and also if the licenses need to be updated.

For information on how to use the floating license, please refer to the user manual for the relevant Workbench (see `http://www.clcbio.com/manuals`).

Plug-ins use the same licensing system as the Workbenches, so all the concepts described here also apply to the plug-in licenses.

3.1 License server set-up on clients

The connection to the license server can be set up as described in the Workbench user manual (see also figure 3.1).

The license server information is stored in a file called `license.properties` in the `settings` folder in the Workbench installation directory. This means, that you need write access to the installation directory (with the default installation directory, you need to be an administrator to have this write access) in order to set up a connection to the license server. The file contains the following:

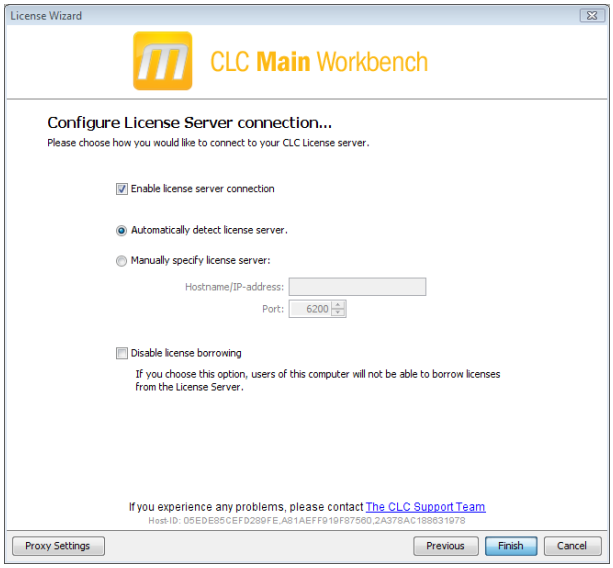
License Wizard
CLC Main Workbench
Configure License Server connection...
Please choose how you would like to connect to your CLC License server.
Enable license server connection
Automatically detect license server.
Manually specify license server:
Hostname/IP-address:
Port:
6200
Disable license borrowing
If you choose this option, users of this computer will not be able to borrow licenses from the License Server.
If you experience any problems, please contact The CLC Support Team
Host-ID: 05EDE85CEFD289FE,A81AEFF919F87560,2A378AC18883197B
Proxy Settings
Previous
Finish
Cancel

Figure 3.1: *Connecting to a license server.*

```
serverip=
serverport=6200
disableborrow=false
autodiscover=true
useserver=true
```

Since all this information is stored in a file in the installation directory, it can easily be copied to all clients, and the license configuration is completed. When the Workbench is started, it will look in this file, and if `useserver=true` then it will try to connect to the license server, and no license dialogs will be shown to the user.

You can download a sample `license.properties` file at http://clcbio.com/files/deployment/license.properties.

Chapter 4

Plug-ins and resources

There is a graphical user interface to install plug-ins called the **Plug-in Manager** ( ) which is invoked in the **Help** menu (see figure 5.1).

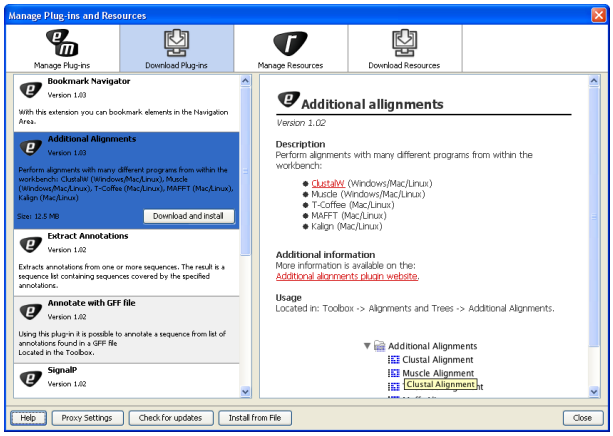

Figure 4.1: *The Plug-in Manager.*

Plug-ins are either general modules or extensions provided by CLC bio (see `http://www.clcbio.com/plugins`) or can be custom-made plug-ins specific to your organization.

Plug-ins can either be downloaded and installed directly in the **Plug-in Manager**, or they can be installed from a file Install from File button at the bottom of the **Plug-in Manager**.

Resources are installed in the same way as plug-ins. Resources can be e.g. PFAM databases used by the Workbench's PFAM Domain Search ( ).

Installing a plug-in is basically just a matter of putting files in the right folder. All plug-in files are put in `plugins` and all resources in `resource` in the installation directory. This means that the contents of these folders can be copied to other computers, and they will have the plug-ins installed.

Licenses for the plug-ins are handled the same way as the Workbench licenses, see section 3.

Chapter 5

Workflows

There is a graphical user interface to install workflows called **Workflows** ( ) which is invoked in the **Help** menu (see figure **??**).

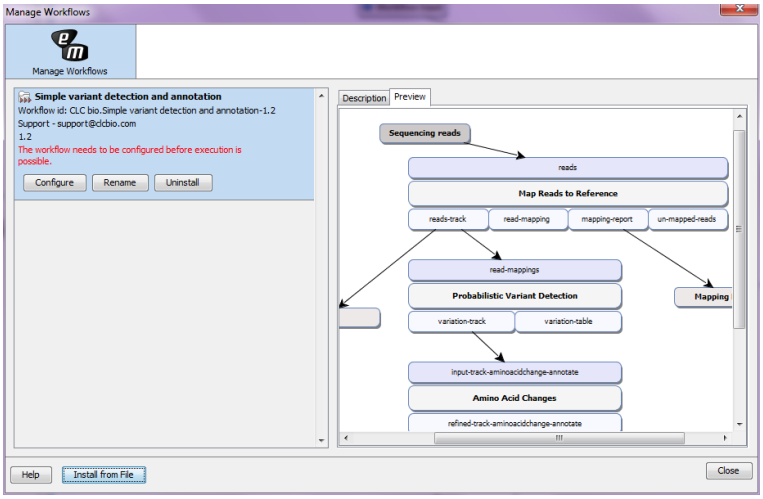

Figure 5.1: *The workflow manager.*

Workflows can be created in any workbench and distributed as an installer file that can be installed in any workbench or server.

A workflow is always installed per user. The workflow definition is stored in the user home (see section 10.2).

The ability to install workflows can be disabled by the policy (see section 7).

When the workbench is part of a *CLC Genomics Server* set-up, it will be a great advantage to manage workflows to be used by all users on the server which means no local deployment when workflows are updated (see more in the user manual at `http://www.clcbio.com/usermanuals`).

Chapter 6

Connecting to a CLC Server

Information about server name and port can be stored in a file called `serverinfo.properties` in the `settings` folder in the Workbench installation directory. When the user opens the log-in dialog, the Workbench will read in the information from this file.

If the file does not exist, the information that the user enters will be saved in the user settings. The user name and password is stored with the user settings.

The `serverinfo.properties` file contains the following:

```
port=7777
host=hostname
```

Since all this information is stored in a file in the installation directory, it can easily be copied to all clients.

You can download a sample `serverinfo.properties` file at http://clcbio.com/files/deployment/serverinfo.properties.

Chapter 7

Security policies

The Workbench has a security policy configuration that enables administrators to restrict users' access to:

- **Tools accessing services on the internet.** This includes NCBI BLAST, NCBI and Uniprot Searches.
- **Notifications about updates.** Update notifications on new Workbench and plug-in versions
- **Plug-in management.** Installation of plug-ins.

The configuration is specified in a simple properties file called `policy.properties` that resides in the `settings` folder of the installation directory (e.g. `C:\Program Files\CLC Main Workbench 6\settings` on Windows). Note that users without administrator access will not be able to change the contents of this file. Each of the following keys can be followed be either *allow* or *deny*:

workbench_version_check Controls whether notifications for Workbench updates should be shown.

plugin_version_check Controls whether notifications for plug-in updates should be shown. Note that if `plugin_download` is not allowed, plug-in update notifications will not be shown, regardless of this setting.

online_ncbi_search Controls whether the **Search for Sequences at NCBI (** **)** in the **Download** menu should be available (this can be used to search for and download sequences from NCBI).

online_structure_search Controls whether the **Search for Structures at NCBI (** **)** in the **Download** menu should be available (this can be used to search for and download 3D structures from NCBI).

online_uniprot_search Controls whether the **Search for Sequences in UniProt (** **)** in the **Download** menu should be available (this can be used to search for and download sequences from Swissprot/UniProt).

online_ncbi_blast Controls whether all the tools performing BLAST at NCBI's servers should be available. This is **NCBI BLAST (** **)** both from the **Toolbox** and from sequence selections.

plugin_manage Controls whether the **Plug-ins and Resources** manager should be available for the user. Note that users can still install plug-in updates if `plugin_download` and `plugin_version_check` are allowed.

plugin_file_install Controls whether this user should be allowed to install plug-ins and resources from a local file.

plugin_download Controls whether this user should be allowed to install CLC plug-ins and resources downloaded directly within the plug-in manager dialog. This also includes manually checking for updated plug-ins in the **Plug-ins and Resources** manager dialog and also the automatic check for plug-in updates at start-up.

workflow_file_install Controls whether the user should be allowed to install workflows from a file.

workflow_manage Controls whether the user should be allowed to manage workflows.

workflow_download Controls whether the user should be allowed to download and install workflows form the CLC workflows repository.

Per default, there is no policy.properties file, so everything is allowed. A commented sample file that you can download and edit is located at `http://clcbio.com/files/deployment/2/policy.properties`. Download the file and place it in the `settings` folder, update the relevant values, and the new policy will take effect next time the Workbench is started.

Chapter 8

Storing and backing up data

This chapter explains how data is stored, gives general guidance on size of data, and outlines configurations needed for running analyses on large amounts of data.

8.1 Storing data

8.1.1 Data structure

The data in the **Navigation Area** is organized into a number of **Locations**. When the the CLC workbenches except the *CLC Genomics Workbench* is started for the first time, there is one location called *CLC_Data* (unless your computer administrator has configured the installation otherwise).

A location represents a folder on the computer: The data shown under a location in the **Navigation Area** is stored on the computer in the folder which the location points to.

This is explained visually in figure 8.1.

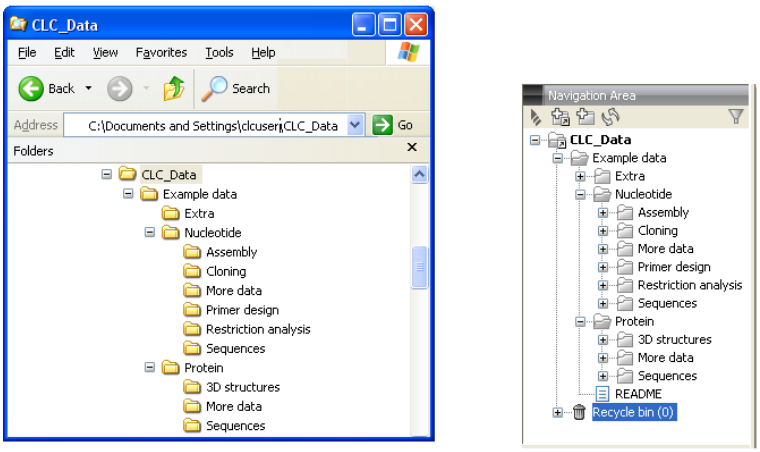

Figure 8.1: *In this example the location called 'CLC_Data' points to the folder at C:\Documents and settings\clcuser\CLC_Data.*

If the Workbench is connected to a CLC Server, the server's locations will automatically show up when the user is logged in. This chapter does not deal with server locations - please refer to the server user manual at http://www.clcbio.com/usermanuals.

The list of locations is stored in a file called `model_settings_300.xml` in the `settings` folder in the user home (see section 10.2). We do not recommend manual editing of this file, although it is standard xml.

8.1.2 Changing the default location

In some set-ups, storing data in the default location `CLC_Data` in the user home is not desired. This could be for roaming user profiles or in situations where there is a quota on this disk.

The default location that is used the first time the Workbench starts can be configured in the `path.properties` file that resides in the `settings` folder of the installation directory (e.g. `C:\Program Files\CLC Main Workbench 5\settings` on Windows). Add a line like this to the file to change the default location: The file should include one line like this:

```
datadir = c:\clcdata
```

The following variables can be used to construct the path to the desired location:

$user the user name of the current user

$home the home directory of the current user

$product the short name of the workbench (example: clcgenomicswb or clcmainwb)

A few examples:

```
datadir = $home/CLC_Data        (default)
datadir = X:\clcstorage\$user   (seperate disk / network mount in Windows)
```

Note that the folder does not need to exist - it will be created if needed. You can use both slash (normally used on Mac and Linux) and backslash (normally used on Windows) in the configuration file - the Workbench will use the appropriate one depending on the platform.

Note that the default location is only considered the very first time the Workbench starts. When the Workbench closes the first time, the `model_settings_300.xml` file is created and this is where it will look for the locations further on. Deleting this file will make the Workbench look for the `datadir` property in the `path.properties` file. The `model_settings_300.xml` file is located in the `settings` folder in the user home (see section 10.2).

You can download a sample `path.properties` file at http://clcbio.com/files/deployment/path.properties.

8.2 Back-up of data

Since all data used in the Workbench is stored as files in the locations specified, a back-up procedure has to include all the locations. If the data needs to be restored from a back-up, simply copy the files back into the folder locations and start the Workbench.

Database locations needs a different back-up procedure.

Besides the data itself, user-level settings should also be included in the back-up (see section 10.2).

8.3 Special configurations for large amounts of data

Especially the *CLC Genomics Workbench* is often used with large amounts of data. This means that special configurations often need to be made. This concerns locations for temporary data and disk space in general.

8.3.1 Temporary data

The Workbench has a built-in cache system that intends to make sure that the Workbench does not run out of memory even for large data sets. During various processes such as assembly and RNA-Seq analysis, the Workbench often writes temporary files to the disk. Depending on the data set, these temporary files can take up a lot of disk space.

If there is not enough space in the default tmp directory, the tmp directory can be re-directed:

Create a text file called `path.properties` and save it in the `settings` folder in the Workbench installation directory. The file should include one line like this:

```
tmpdir = /path-to-temp
```

Instead of path-to-temp you write the absolute path to the new tmp directory. When the Workbench is restarted, it will then use the new directory for storing temporary data.

You can download a sample `path.properties` file at http://clcbio.com/files/deployment/path.properties.

Note! It is imperative for acceptable performance that data transfer to the temp directory is not over a network connection. Since the Workbench will spend a lot of time writing and reading these files, disk speed has a great impact on overall performance when working with large data sets.

8.3.2 Disk space requirements

It is hard to give general guidance on disk space requirements, but we have made an example of a typical work flow for *CLC Genomics Workbench* to illustrate.

For calculating disk space for next-generation sequencing data you need to consider the following:

- Reads are imported and take up space as raw reads (see details below). Once imported, you can delete the original sequence file if you do not need it for other purposes.
- When the data has been assembled, either de-novo or against a reference, they take up space once again (this time more space since there is also information about where they map etc).
- Reference sequences also take up space.
- The computer doing the analysis needs space for tmp files. Once the assembly is done, the temporary files are deleted. The temporary files usually do not take up more space than the final result

The formulae giving the disk space usage:

Bytes per read: 28 + (length of read name) + 0.25 x (length of read)

Note that you can discard read names during import.

If quality scores are present, add: 6 + (length of read)

If color space encoding is present, add: 7

As an example, a data set of 5.2 million 35 bp reads imported by *CLC Genomics Workbench* using the **Discard sequence names** option including quality scores takes up:

5,244,764 x ((28 + 0 + 0.25 x 35) + (6 + 35)) = 389 MB

When mapped to a 4.7 Mbp annotated reference sequence, the mapping results takes up 473 MB.

Chapter 9

System resources

You can specify the amount of system resources that the Workbench is allowed to use. This can be done for both memory and CPU.

9.1 Setting the amount of memory available for the JVM

When running the Workbench, the Java Virtual Machine (JVM) needs to know how much memory it can use. This depends on the amount of physical memory (RAM) and can thus be different from computer to computer. Therefore, the installer investigates the amount of RAM during installation and sets the amount of memory that the JVM can use when running the Workbench.

On **Windows** and **Linux**, this value is stored in a property file called `workbenchname.vmoptions` (e.g. `clcmainwb.vmoptions`) which contains a text like this:

```
-Xmx1400m
```

The number (1400) is the amount of memory the Workbench is allowed to use. The file is placed in the installation folder (e.g. `C:\Program Files\CLC Main Workbench 6\clcmainwb.vmoptions`).

On **Mac OS X**, the -Xmx value is stored in `Info.plist` in the application bundle (Control-click the application and choose "Show Package Contents").

The value is set to 50% of the computers RAM per default, and at a maximum of 1400 MB for 32-bit systems and a maximum at 50GB for 64-bit systems (at this point there is no performance increase by setting it higher because of overheads in memory management). If you do not wish to use the installer on each computer and use an image instead, either make sure all computers have the same amount of RAM, or set the number to 50% of the computer with the smallest amount of RAM (this value should not be lower than 200 MB, and for genomics-scale data, it should be significantly higher).

You can manually change the contents if you want to raise or lower this value. In case you get an out of memory error you may need to allocate a higher amount of memory by increasing the number in the .vmoptions file. In case you get a java exception saying "The JVM could not be started. The main method may have thrown an exception" you may need to decrease the value in the .vmoptions file.

9.2 Setting the number of cores to use

A number of the algorithms in the *CLC Genomics Workbench* , in the case of large jobs, use all the cores available on your system to make the analysis as fast as possible.

If you wish to restrict this to a predefined number of cores, this can be done with a properties file: Create a text file called `cpu.properties` and save it in the `settings` folder in the Workbench installation directory. The file should include one line like this:

```
maxcores = 1
```

Instead of `1` you write the maximum number of cores that the Workbench is allowed to use. When the Workbench is restarted, it will comply with this setting. Please note that this is not a guarantee that the Workbench will never use more, but that will be for very brief and infrequent peaks and should not affect performance of other applications running on your system. The only exception is when a user starts several jobs to run simultaneously.

You can download a sample `cpu.properties` file at http://clcbio.com/files/deployment/cpu.properties.

Chapter 10

Overview - where do we put things?

This part gives you an overview of all the parts of a CLC Workbench installation. Some parts are at the level of the computer, i.e. shared by all users, whereas other parts are at the user level.

10.1 Computer-level information

In the *Workbench installation directory*, you will find the following:

Licenses The license information depends on what kind of license you use:

- **Fixed licenses** Stored in the `licenses` folder (they are unique for each computer)
- **Floating licenses** Information about the license server connection is stored in the `license.propertie` file in the `settings` folder.

Plug-ins are stored in the `plug-ins` folder and can be copied to other computers running the same version of the Workbench.

Resources are stored in the `resource` folder and can be copied to other computers running the same version of the Workbench.

Security policy is stored in a properties file in the `settings` folder

Path for default location and temporary data is stored in a properties file in the `settings` folder

Information on server port and host is stored in a properties file in the `settings` folder

Memory allocation for the VM is stored in the `workbenchname.vmoptions` file.

Proxy settings Information about proxy server (when the Workbench needs access to online services) is stored in the `proxy.properties` file in the `settings` folder (the file will only be created if a proxy server has been specified).

In addition, file associations for .clc files are stored in the registry database on Windows.

10.2 User-level information

The user-level information is found in the application data folder:

Windows 2000 and XP `C:\Documents and settings\username\Application data\CLC bio`

Windows Vista and 7 `C:\Users\username\Appdata\Roaming\CLC bio`

Mac OS X `User home/Library/Application Support/CLC bio`

Linux `$HOME/.clcbio`

In this folder, the following information may be useful to you:

User settings The user settings file stores information such as view settings, parameters, workspaces, user name and password to server log-ins and other settings customized by the user. The user settings file is found in the application data folder under `settings/workbench name/version name`. It is recommended to back up the user settings file. When upgrading to a new version of the Workbench, the user's old user settings file is copied by the Workbench the first time it is run.

Locations Information about which locations the user has added in the **Navigation Area** is stored in the `model_settings_300.xml` file in the `settings` folder.

Error logs to be sent to support@clcbio.com for use in case of program errors are `output.log` and `error.log` and they are found in the `log` folder.

Workflows Workflows are saved in a `workflows/workbench name/version name` folder. When upgrading to a new version of the Workbench, the user's old workflows are copied by the Workbench the first time it is run, unless there are major changes to the workflow framework that breaks compatibility.

Index

CLC Bioinformatics Database, 5
CLC Server, 5
32-bit installer, 8
64 bit installer, 8

Back up
    user settings, 25
Back-up, 19
Block access to internet, 16

Command-line installation, 10
Cores, restrict usage, 23
CPU, restrict usage of, 23

Data storage, 18
Data structure, 18
Database
    local, 18

Error log, 25

Floating license, 11

GHOST image, 6

Image, copy, 6
Install
    plug-ins, 13
    resources, 13
    workflows, 14
Installers, overview, 9
Introduction, 5

Java, 8
JRE, Java Runtime Environment, 8
JVM, Java Virtual Machine, 8

License order ID, 11
License server, 11
    set-up on clients, 11
Licenses, 11
    for plug-ins, 11
Linux, 5
Linux installer vs. package, 8
Locations, 25

Memory allocation, 22

Online tools, block access to, 16
Output log, 25

Plug-in licenses, 11
Plug-in Manager, 13
Plug-ins, 13
Policy, 16
Proxy server, 24

Quiet installation, 10

RAM, 22
Resources, 13
.rpm, Linux package, 8

Security policies, 16
Server connection, 15
.sh, Linux installer, 8
Silent installation, 10
System requirements, 7
System resources, 22

Temporary data, 20

User settings, 25

VM, Virtual Machine, 8
.vmoptions, memory allocation, 22

Workflows, 14

Xmx argument, 22